

# Weekly report

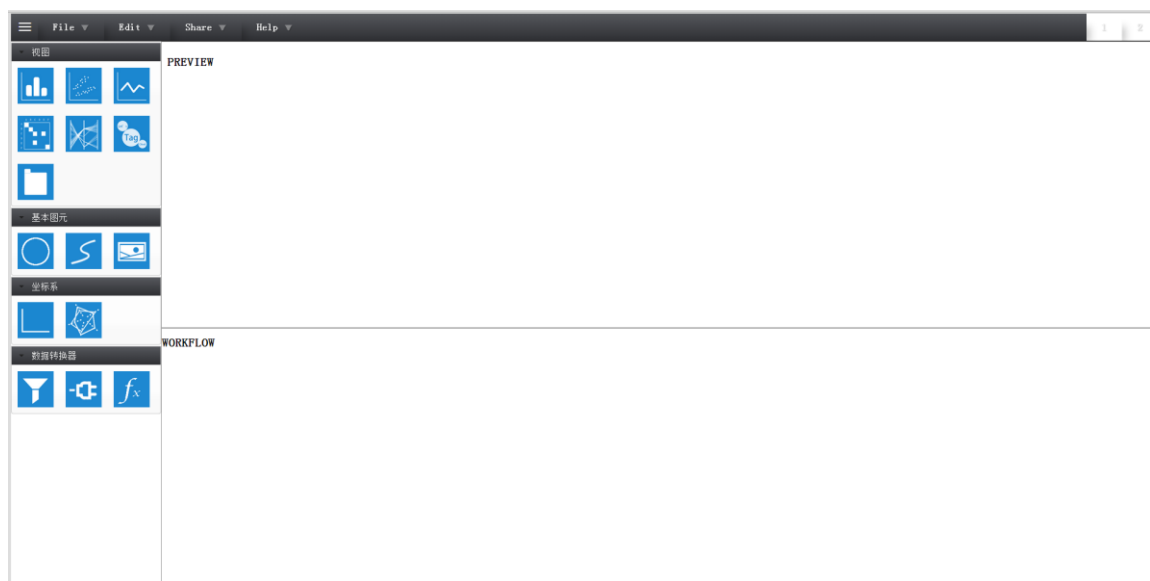
2014.09.22~2014.09.28

胡万祺

## 一、本周工作总结

Visketch:

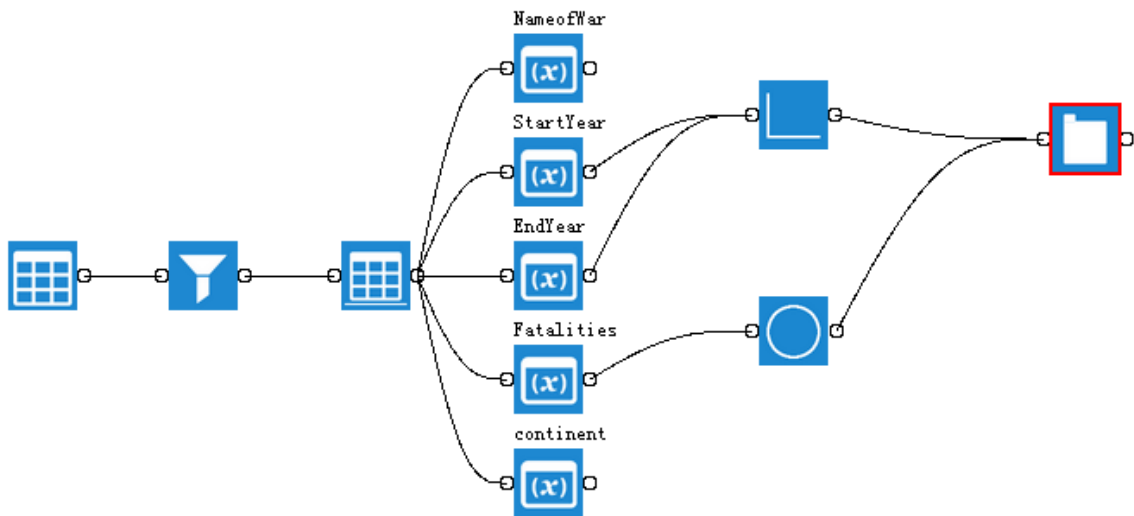
### 1、新界面的整合。



右上角的 1 和 2 负责控制 preview 窗口和 workflow 窗口的开关。

### 2、实现 workflow 用有向链接图的形式储存。

我们在使用 visketch 构建一个可视化的完整流程大概可以用如下图这样一个 workflow 结构来呈现。



之前实现的代码中只保存了 workflow 中每个节点的信息以及两两节点间的连接信息。后来发现这样的信息是不够的。因为当我们已经构建好了视图，然后需要再去对原始数据做处理的时候，已经与数据变量绑定好了的视图应该是要能根据数据的变换而做出变化的。要让图元模块响应绑定的数据的变化而变化，必须要有个能记录 workflow 各个层次的具体信息的数据结构。和沈晗晓、小马哥讨论之后决定采用有向链接图的形式。代码截图如下

```

10  function Node() {
11      this.cid = '';
12      this.level = '';
13      this.predecessor = [];
14      this.descendant = [];
15  };
16
17  function Node(cid, predecessor, descendant) {
18      this.cid = cid;
19      this.level = '';
20      this.predecessor = predecessor;
21      this.descendant = descendant;
22  };
23
24  Node.prototype.setDescendant = function (descendant) {
25      this.descendant.push(descendant);
26  };
27
28  Node.prototype.getCid = function () {
29      return this.cid;
30  };
31
32  Node.prototype.getDescendant = function () {
33      return this.descendant;
34  };
35
36  Node.prototype.getPredecessor = function () {
37      return this.predecessor;
38  };

```

```

59 function Graph() {
60     this.nodes = [];
61 };
62
63 Graph.prototype.findNode = function(cid){
64     for(var i = 0; i < this.nodes.length; i++){
65         if(this.nodes[i].getCid() == cid){
66             return i;
67         }
68     }
69     return -1;
70 };
71
72 Graph.prototype.update = function(cid){
73     var index = this.findNode(cid);
74     this.nodes[index].level = 500 + this.nodes[index].cid;
75     //
76     var descendants = this.nodes[index].getDescendant();
77     console.log(descendants);
78     if (descendants.length != 0) {
79         for (var i = 0; i < descendants.length; i++) {
80             var index = this.findNode(descendants[i]);
81             this.update(this.nodes[index].getCid());
82         }
83     }
84 };
85
86 Graph.prototype.addNode = function (cid, PNode) {
87     var index = -1;
88     for (var i = 0; i < PNode.length; i++) {
89         index = this.findNode(PNode[i]);
90         this.nodes[index].setDescendant(cid);
91     }
92     var node = new Node(cid, PNode, []);
93     this.nodes.push(node);
94 };

```

## 二、下周工作展望

- 1、完成除可编程模块展示的所有功能，包括高亮、交互等。
- 2、与劳天溢合作将 NBA 可视化视图接入系统。